Jeff Hilland
draft-hilland-iwarp-verbs-v1.0 Hewlett-Packard Company
Paul Culley
Hewlett-Packard Company
Jim Pinkerton
Microsoft Corporation
Renato Recio
IBM Corporation

April, 2003

RDMA Protocol Verbs Specification (Version 1.0)

1 Status of this Memo

This document is a Release Specification of the RDMA Consortium. Copies of this document and associated errata may be found at http://www.rdmaconsortium.org.

2 Abstract

This document describes an abstract interface to a RDMA enabled NIC (RNIC). This interface is implemented as a combination of the RNIC, its associated firmware, and host software. It provides access to the RNIC queuing and memory management resources, as well as the underlying networking layers.

Block List - A list of physical addresses describing a set of memory blocks, which specifies the block size, list of physical addresses, and offset to the start of the memory region of the first block. Each block has the same length and that length can be any value in the range supported by the RNIC. Each block may start at a byte granularity address. The starting address for the entire list may be an offset into the first block and the entire list may have any length.

Complete (Completed, Completion, Completes) - When the Consumer can determine that a particular RDMA Operation has performed all functions specified for the RDMA Operation, including Placement and Delivery. This can be determined through a Work Completion for Signaled Work Requests. For Unsignaled Work Requests, this means that the Completion Rules have been met. Note that this is a superset of the [RDMAP] definition for RDMA Completion.

Completion Error - A Processing Error reported through the Completion Queue.

Completion Queue (CQ) - A sharable queue containing one or more entries which can contain Completion Queue Entries. A CQ is used to create a single point of completion notification for multiple Work Queues. The Work Queues associated with a Completion Queue may be from different QPs and of differing queue types (SQs or RQs).

Completion Queue Entry (CQE) - The RNIC Interface internal representation of a Work Completion.

Completion Status - The resultant status of a Work Request returned as part of a Work Completion.

Consumer, Verbs Consumer - A software process that communicates using RDMA/DDP Verbs. The Consumer typically consists of an application program, or an operating system adaptation layer, which provides some OS specific API.

Direct Data Placement Protocol (DDP) - A wire protocol that supports Direct Data Placement by associating explicit memory buffer placement information with the LLP payload units.

Data Delivery (Delivery, Delivered, Delivers) - Delivery is defined as the process of informing the ULP or Consumer that a particular Message is available for use. This is specifically different from Data Placement, which may generally occur in any order, while the order of Data Delivery is strictly defined.

Data Placement (Placement, Placed, Places) - A mechanism whereby ULP data contained within RDMA/DDP Segments may be put directly into

Remote Peer - The RDMA protocol implementation on the opposite end of the connection. Used to refer to the remote entity when describing protocol exchanges or other interactions between two Nodes.

Remote RDMA Read Operation - a sequence of events that begins upon receipt of an incoming RDMA Read Request by the RI and stays in-process until the corresponding RDMA Read Response Message has been generated. This includes posting the RDMA Read Request to the Inbound RDMA Read Request Queue (See Section 6.5 - Outstanding RDMA Read Resource Management).

RNIC Interface (RI) - The presentation of the RNIC to the Verbs Consumer as implemented through the combination of the RNIC and the RNIC device driver.

Scatter/Gather Element (SGE) - An individual entry in a Scatter/Gather List. Each SGE consists of an STag, Tagged Offset and Length.

Scatter/Gather List (SGL) - A List of Scatter/Gather Elements. The list describes one or more ULP Buffers which will have their data gathered on transmission or scattered upon reception.

Send - An RDMA Operation that transfers the contents of an Untagged buffer from the Local Peer to an Untagged buffer at the Remote Peer.

Send Operation Types - The set of Send operations that result in the consumption of a Receive Queue Work Request at the Data Sink. Specifically this includes Send, Send with Invalidate, Send with Solicited Event and Send with Solicited Event & Invalidate.

Send Queue (SQ) - One of the two Work Queues associated with a Queue Pair. The Send Queue contains PostSQ Work Queue Elements that have specific operation types, such as Send Type, RDMA Write, or RDMA Read Type Operations, as well as STag operations such as Bind and Invalidate.

Shared Memory Region - An MR that currently shares; or at one time shared, the Physical Buffer List associated with the Memory Region. Specifically, the PBL is currently shared or was previously shared with another Memory Region.

Shared Receive Queue - An optional mechanism which allows the Receive Queues from multiple QPs to retrieve Receive Queue Work Queue Elements from the same shared queue as needed.

Signaled - A WR which requires that the RNIC generate a Work Completion.

Solicited Event (SE) - A facility by which an RDMA Operation sender may cause an Event to be generated at the recipient, if the recipient is configured to generate such an Event, when a Send with Solicited Event or Send with Solicited Event & Invalidate Message is received.

Steering Tag (STag) - An identifier of a Memory Window or Memory Region. STags are composed of two components: an STag Index and an STag Key. The Consumer forms the STag by combining the STag Index with the STag Key. This specification further refines the definitions of STags contained in [RDMAP] and [DDP].

STag Key - The least significant 8 bit portion of an STag. This field of an STag can be set to any value by the Consumer when performing a Memory Registration operation, such as Bind Memory Window, Fast-Register Memory Region and Register Memory Region.

STag Index - The most significant 24 bits of an STag. This field of the STag is managed by the RI and is treated as an opaque object by the Consumer.

Tagged Buffer - A buffer that can be Advertised to a Remote Peer through exchange of an STag, Tagged Offset, and length.

Tagged Offset (TO) - The offset within a Tagged Buffer.

Terminate - An RDMA Message used by a Node to pass an error indication to the Remote Peer on an RDMA Stream.

Upper Layer Protocol (ULP) - The protocol layer above the Verb layer. An example is SDP.

ULP Buffer - A buffer owned above the RI that can be represented within the RNIC, in whole or in part, by a Memory Window or a Memory Region.

ULP Message - The ULP data that is handed to a specific protocol layer for transmission. Data boundaries are preserved as they are transmitted through iWARP.

ULP Payload - The portion of a ULP Message that is contained within a single protocol segment or packet (e.g. a DDP Segment).

Unaffiliated Asynchronous Event - This is an indication from the Verb layer to the Consumer that an event has occurred unrelated to any single identifiable RNIC Resource.

Unsignaled - A Work Request which only generates a Work Completion if it encounters an error during processing.

which help a Consumer to efficiently notice when WRs have completed processing in the RI. There may be thousands of CQs per RNIC.

Event Handlers provide the mechanism for Consumers to be notified of Asynchronous Events which occur within the RI but which cannot be reported through the Completion Queues due to their asynchronous nature or the fact that they are not easily associated with a Work Completion.

5.1 The RNIC

Consumers gain access to an RNIC through the RNIC Interface. The Verbs allow the Consumer to open the RNIC, retrieve RNIC attributes, and close the RNIC.

All resources MUST be in the scope of the RNIC on which they are created. This means that there is no requirement for resources on one RNIC to be available, associated with or meaningful to another RNIC, even if they are managed by the same RNIC driver. This includes all QPs, STags, PDs, CQs, and multiple Completion Event Handlers. This also means that any IDs which are created by the RI are specific to that RNIC and are not guaranteed to be unique across all RNICs.

An intent of the architecture is to allow an implementation to pass Work Requests and Work Completions to and from a Non-Privileged Mode Consumer process directly to and from the RNIC. Another intent of the architecture is to optimize for a Privileged Mode implementation, which shares the Work Request and Work Completion requirements of Non-Privileged Mode Consumers but has slightly different memory management requirements.

Because the architecture attempts to optimize for both Privileged Mode and Non-Privileged Mode Consumers, there are some Verbs and Verb modes which are not allowed to be executed by non-Privileged Mode Consumers. An example of this is the use of the STag of zero or the ability to do Fast-Register WRs. In addition, there are some operations that, while being allowed in kernel mode, are intended to be used by Non-Privileged mode applications. An example of this is Memory Windows. Any restrictions are clearly specified in this document where required.

5.1.1 RNIC Resources

RNIC Resources can be allocated from a variety of places. They can be allocated in host memory on behalf of the Consumer or allocated within the RNIC. Where an RNIC allocates resources is implementation specific. Consequently, values that the RNIC returns as output modifiers when Querying the RNIC indicate the maximum amount of any

8 least significant bits of the STag. The STag Index is the 24 most significant bits of the STag.

The 8 bit STag Key is provided by the Consumer. The Consumer can use the STag Key in any way it desires. For example, it can be used as an incrementing value to help discover application errors by using a different value with each registration. As a general rule, the Consumer provides the STag Key to the RI whenever the consumer causes the transition of an STag to the Valid state, or when the STag is being Invalidated. In the Invalid state, only the STag Index is meaningful.

There is no default value for the STag Key. The RI MUST use the STag Key provided by the Consumer for the following Verbs:

* Register Non-Shared Memory Region,

* Register Shared Memory Region,

* Reregister Non-Shared Memory Region,

* PostSQ Verb Fast-Register Non-Shared Memory Region operation, and

* PostSQ Verb Bind operation,

* PostSQ Invalidate Local STag.

The RI MUST return the value of the STag Index sub-field on an invocation of the following:

* Allocate Non-Shared Memory Region STag,

* Allocate Memory Window,

* Register Non-Shared Memory Region,

* Register Shared Memory Region, and

* Reregister Non-Shared Memory Region.

The RI MUST use the same STag Index sub-field as was passed in by the Consumer, on an invocation of the following:

* Query Memory Region,

* Query Memory Window,

* Register Shared Memory Region,

- The memory access as specified by the TO & length is within the base and bounds of the Memory Region. The RI MUST enforce this with a byte level granularity.

If the length of the access is zero, the RI MUST NOT perform any of the above checks on the Memory Region.

7.7 Querying Memory Regions

Memory Regions have attributes that can be retrieved through the Query Memory Region Verb. The RI MUST support the complete list of QP attributes as described in Section 9.2.6.3 - Query Memory Region.

7.8 Invalidating Memory Regions

When access to a Non-Shared Memory Region by an RI is no longer required, but the Consumer wants to retain the STag for use in future Fast-Register Non-Shared Memory Region and RI-Reregister Non-Shared Memory Region Verb invocations, the Consumer may directly invalidate access to the Non-Shared Memory Region through an Invalidate Local STag WR or an RDMA Read with Invalidate Local STag WR. Additionally, an STag may be invalidated by a remote Consumer through the use of a Send with Invalidate Message or a Send with Solicited Event and Invalidate Message.

Multiple Memory Regions can represent memory locations that have been registered multiple times. The invalidation of a single STag prevents RNIC access to those memory locations via the STag associated with that Memory Region. Access to the memory locations via STags associated with other Memory Regions other than the STag being Invalidated MUST NOT be affected. Invalidating an STag associated with a Memory Region that partially or completely overlap other Memory Regions MUST NOT cause the RI to affect the registration of those other Memory Regions.

The requirements for unpinning the physical buffers associated with deallocated Memory Regions are covered in Section 7.6.2 - Physical Buffer Lists.

Invalidating an STag associated with a Shared Memory Region MUST result in an Completion Error. Consequently, using an STag associated with a Shared Memory Region under the following conditions will cause a Completion Error at the Data Sink that results in the LLP Stream being torn down after the data transfer operation takes place:

- As the STag specified in an Invalidate Local STag WR.

- As the Data Sink STag for an RDMA Read with Invalidate Local STag WR.

draft-recio-iwarp-rdmap-v1.0

R. Recio
IBM Corporation
P. Culley
Hewlett-Packard Company
D. Garcia
Hewlett-Packard Company
J. Hilland
Hewlett-Packard Company

October 2002

An RDMA Protocol Specification (Version 1.0)

1 Status of this Memo

This document is a Release Specification of the RDMA Consortium. Copies of this document and associated errata may be found at http://www.rdmaconsortium.org.

2 Abstract

This document defines a Remote Direct Memory Access Protocol (RDMAP) that operates over the Direct Data Placement Protocol (DDP protocol). RDMAP provides read and write services directly to applications and enables data to be transferred directly into ULP Buffers without intermediate data copies. It also enables a kernel bypass implementation.

6 Header Format

The control information of RDMA Messages is included in DDP protocol defined header fields, with the following exceptions:

* The first octet reserved for ULP usage on all DDP Messages in the DDP Protocol (i.e. the RsvdULP Field) is used by RDMAP to carry the RDMA Message Opcode and the RDMAP version. This octet is known as the RDMAP Control Field in this specification. For Send with Invalidate and Send with Solicited Event and Invalidate, RDMAP uses the second through fifth octets provided by DDP on Untagged DDP Messages to carry the STag that will be Invalidated.

* The RDMA Message length is passed by the RDMAP layer to the DDP layer on all outbound transfers.

* For RDMA Read Request Messages, the RDMA Read Message Size is included in the RDMA Read Request Header.

* The RDMA Message length is passed to the RDMAP Layer by the DDP layer on inbound Untagged Buffer transfers.

* Two RDMA Messages carry additional RDMAP headers. The RDMA Read Request carries the Data Sink and Data Source buffer descriptions, including buffer length. The Terminate carries additional information associated with the error that caused the Terminate.

6.1 RDMAP Control and Invalidate STag Field

The version of RDMAP defined by this specification uses all 8 bits of the RDMAP Control Field. The first octet reserved for ULP use in the DDP Protocol MUST be used by the RDMAP to carry the RDMAP Control Field. The ordering of the bits in the first octet MUST be as defined in Figure 3 DDP Control, RDMAP Control, and Invalidate STag Field. For Send with Invalidate and Send with Solicited Event and Invalidate, the second through fifth octets of the DDP RsvdULP field MUST be used by RDMAP to carry the Invalidate STag. Figure 3 DDP Control, RDMAP Control, and Invalidate STag Field depicts the format of the DDP Control and RDMAP Control fields. (Note: In Figure 3 DDP Control, RDMAP Control, and Invalidate STag Field, the DDP Header is offset by 16 bits to accommodate the MPA header defined in [MPA]. The MPA header is only present if DDP is layered on top of MPA.)